UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/761,163	01/20/2004	Francisco Dc La Cruz	AUS920040005US1	2564

43307 7590 09/28/2006

IBM CORP (AP)
C/O AMY PATTILLO
P. O. BOX 161327
AUSTIN, TX 78716

EXAMINER	
URICK, MATTHEW T	

ART UNIT	PAPER NUMBER
2113	

DATE MAILED: 09/28/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/761,163	CRUZ ET AL.	
	Examiner	Art Unit	
	Matt Urick	2113	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *1/20/04*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-33* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☒ Claim(s) *13* is/are allowed.

6)☒ Claim(s) *1-12 and 14-33* is/are rejected.

7)☒ Claim(s) *9* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *1/20/04* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1, 3-8, 10-11, 15, 17-23, 25, and 27-32 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gadir (US Patent 2003/0018927 A1) in view of Cramer (US Patent 6,920,580).

As per claim 1, Gadir discloses:

A system for managing a high availability cluster during failover, comprising:

a primary node running a middleware stack for supporting web applications, wherein a plurality of layers of said middleware stack are active, wherein said primary node is assigned a virtual IP address to which requests are directed (¶ 10: two or more nodes contain one or more virtual servers [layers], controlling resources and file systems. One node is the "master" node, while the rest are "slave" nodes, ¶ 13: each node is assigned an IP address);

a secondary node running a redundant middleware stack mirroring said plurality of layers of said middleware stack of said primary node (¶ 10),

wherein a first selection of said plurality of layers of said redundant middleware stack are active and a second selection of said plurality of layers of said redundant middleware stack are in standby (¶ 10);

a data replication partition shared between said primary node and said secondary node with data accessible to a selection of said plurality of layers of said active middleware stack (¶ 29 lines 12-15),

wherein said selection of said plurality of layers of said active middleware stack correspond to said second selection of said plurality of layers of said redundant middleware stack in standby (¶ 4: both nodes may run identical software with identical configurations); and

a heartbeat monitor for detecting a failure at said primary node, wherein in response to detecting said failure at said primary node, said heartbeat monitor transfers said virtual IP address from said primary node to said secondary node (¶ 38 lines 7-15),

said heartbeat monitor remounts said data replication partition for access by said secondary node (¶ 6),

said heartbeat monitor activates said second selection of said plurality of layers of said redundant middleware stack requiring access to said data within said data replication partition (¶ 6).

Gadir does not disclose:

said heartbeat monitor turn off power to said primary node,

Cramer discloses a cluster computing system (figure 1) in which a secondary node may force a takeover of a primary node if the primary node is experiencing a fault (flowchart, figure 5). The secondary node sends a confirmation signal (step 513), and shuts down the primary node if it is not able to do so itself (step 519). Cramer discloses that this prevents resources from being used needlessly by the failed node after the second node has already taken over (column 2 lines 46-55). Gadir discloses that failover causes a disruption of service because of limited availability of services being available, and that lower priority services are eliminated when failover occurs to prevent disruptions (¶ 17). Using the negotiated shutdown system of Cramer would prevent the faulty node from distrupting the system by attempting to access resources after a secondary node has already taken over. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate the negotiated shutdown system of Cramer into the cluster computing system of Gadir, preventing loss of performance during failover.

As per claim 3, Gadir discloses:

The system according to claim 1 for managing a high availability cluster, wherein said middleware stack and said redundant middleware stack comprise said plurality of layers comprising a selection from among a load balancer (¶ 12 last 3 lines), a web server (¶ 8 lines 5-7), a web application server (¶ 8 lines 5-7), a messaging server (¶ 1),

a monitoring server (¶ 5 lines 1-2), and a database control server (¶ 1).

As per claim 4, Gadir discloses:

The system according to claim 1 for managing a high availability cluster, wherein said standby selection of said plurality of layers of said redundant middleware comprise at least one from among a messaging server and a database control server (¶ 1).

As per claim 5, Gadir discloses:

The system according to claim 1 for managing a high availability cluster, wherein a web application server active on said secondary node before failover points to a virtual IP address of a messaging server and a database control server from said middleware stack of said primary node (¶ 32: slave nodes sent status information to master node through the network).

As per claim 6, Gadir discloses:

The system according to claim 5 for managing a high availability cluster, wherein responsive to detecting said failure at said primary node, said heartbeat monitor takes over said virtual IP address and configures said second selection of said plurality of layers of said redundant middleware stack with said virtual IP address on startup (¶ 38 lines 7-15: the virtual IP address is switched as soon as the other virtual servers become active).

As per claim 7, Gadir discloses:

The system according to claim 1 for managing a high availability cluster, wherein a message queue and a relational database are launched in said data replication partition (¶ 29 lines 12-15).

As per claim 8, Gadir discloses:

The system according to claim 1 for managing a high availability cluster, wherein said data replication partition is a Data Replication Block Device (¶ 29 lines 12-15: scribble disk is a Data Replication Block Device).

As per claim 10, Gadir discloses:

The system according to claim 1 for managing a high availability cluster, further comprising:

a service monitoring daemon for monitoring a status of a plurality of services provided by said middleware stack, wherein responsive to detecting a failure of a particular service from among said plurality of services, said service monitoring daemon restarts another instance of said particular service at said primary node with a particular process identifier (¶ 16: ports [service[are replaced when one is faulty, ¶ 38 lines 1-15: each port has a unique identification).

As per claim 11, Gadir discloses:

The system according to claim 1 for managing a high availability cluster, further comprising:

a cross-over cable connected between said primary node and said secondary node via network adapters (¶ 35); and

said heartbeat monitor for sending heartbeat requests via said cross-over cable, wherein in response to not receive a return of a heartbeat request, said heartbeat monitor detects a failure in the non-responsive node (¶ 35).

As per claim 15, Gadir discloses:

A method for managing a high availability cluster during failover, comprising:

controlling a primary node running a middleware stack for supporting web applications, wherein a plurality of layers of said middleware stack are active, wherein said primary node is assigned a virtual IP address to which requests are directed;

controlling a secondary node running a redundant middleware stack mirroring said plurality of layers of said middleware stack of said primary node, wherein a first selection of said plurality of layers of said redundant middleware stack are active and a second selection of said plurality of layers of said redundant middleware stack are in standby (¶ 10: two or more nodes contain one or more virtual servers [layers], controlling resources and file systems. One node is the "master" node, while the rest are "slave" nodes, ¶ 13: each node is assigned an IP address);

managing a data replication partition shared between said primary node and said secondary node with data accessible to a selection of said plurality of layers of-said active middleware stack, wherein said selection of said plurality of layers of said active middleware stack correspond to said second selection of said plurality of layers of said redundant middleware stack in standby (¶ 29 lines 12-15); and

responsive to detecting a failure at said primary node, transferring said virtual IP address from said primary node to said secondary node, remounting said data replication partition for access by said secondary node, activating said second selection of said plurality of layers of said redundant middleware stack which require access to said data within said data replication partition (¶ 38 lines 7-15, ¶ 6).

Gadir does not disclose:

turning turn off power to said primary node

Cramer discloses a cluster computing system (figure 1) in which a secondary node may force a shutdown of a primary node if the primary node is experiencing a fault (flowchart, figure 5). The secondary node sends a confirmation signal (step 513), and shuts down the primary node if it is not able to do so itself (step 519). Cramer discloses that this prevents resources from being used needlessly by the failed node after the second node has already taken over (column 2 lines 46-55). Gadir discloses that failover causes a disruption of service because of limited availability of services being available, and that lower priority services are eliminated when failover occurs to prevent disruptions (¶ 17). Using the negotiated shutdown system of Cramer would prevent the

faulty node from distrupting the system by attempting to access resources after a secondary node has already taken over. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate the negotiated shutdown system of Cramer into the cluster computing system of Gadir, preventing loss of performance during failover.

As per claim 17, Gadir discloses:

The method according to claim 15 for managing a high availability cluster, wherein said middleware stack and said redundant middleware stack comprise said plurality of layers comprising a selection from among a load balancer (¶ 12 last 3 lines), a web server (¶ 8 lines 5-17), a web application server (¶ 8 lines 5-7), a messaging server (¶ 1), a monitoring server (¶ 5 lines 1-2), and a database control server (¶ 1).

As per claim 18, Gadir discloses:

The method according to claim 15 for managing a high availability cluster, wherein said standby selection of said plurality of layers of said redundant middleware comprise at least one from among a messaging server and a database control server (¶ 1).

As per claim 19, Gadir discloses:

The method according to claim 15 for managing a high availability cluster, further comprising: enabling a web application server active on said secondary node before failover points to access a messaging server and a database control server from said middleware stack of said primary node (¶ 32: slave nodes sent status information to master node through the network).

As per claim 20, Gadir discloses:

The method according to claim 15 for managing a high availability cluster, further comprising:

launching a message queue and a relational database in said data replication partition (¶ 29 lines 12-15).

As per claim 21, Gadir discloses:

The method according to claim 15 for managing a high availability cluster, further comprising: responsive to detecting said failure at said primary node, for sending a shutoff command to turn off power to said primary node.

Cramer discloses a cluster computing system (figure 1) in which a secondary node may force a shutdown of a primary node if the primary node is experiencing a fault (flowchart, figure 5). The secondary node sends a confirmation signal (step 513), and shuts down the primary node if it is not able to do so itself (step 519). Cramer discloses that this prevents resources from being used needlessly by the failed node after the second node has already taken over (column 2 lines 46-55). Gadir discloses that

failover causes a disruption of service because of limited availability of services being available, and that lower priority services are eliminated when failover occurs to prevent disruptions (¶ 17). Using the negotiated shutdown system of Cramer would prevent the faulty node from distrupting the system by attempting to access resources after a secondary node has already taken over. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate the negotiated shutdown system of Cramer into the cluster computing system of Gadir, preventing loss of performance during failover.

As per claim 22, Gadir discloses:

The method according to claim 15 for managing a high availability cluster, further comprising:

monitoring a status of a plurality of services provided by said middleware stack;

responsive to detecting a failure of a particular service from among said plurality of services, restarting another instance of said particular service at said primary node with a particular persistent identifier (¶ 16: ports [service] are replaced when one is faulty, ¶ 38 lines 1-15: each port has a unique identification).

As per claim 23, Gadir discloses:

The method according to claim 15 for managing a high availability cluster, further comprising:

sending heartbeat requests between said primary node and said secondary node (¶ 28 lines 7-15, ¶ 6); and

responsive to not receive a return of a heartbeat request, detecting a failure in the non-responsive node (¶ 28 lines 7-15, ¶ 6).

As per claim 25, Gadir discloses:

A computer program product, residing on a computer readable medium, for managing a high availability cluster during failover, comprising:

means for controlling a primary node running a middleware stack for supporting web applications, wherein a plurality of layers of said middleware stack are active, wherein said primary node is assigned a virtual IP address to which requests are directed (¶ 10: two or more nodes contain one or more virtual servers [layers], controlling resources and file systems. One node is the "master" node, while the rest are "slave" nodes, ¶ 13: each node is assigned an IP address);

means for controlling a secondary node running a redundant middleware stack mirroring said plurality of layers of said middleware stack of said primary node, wherein a first selection of said plurality of layers of said redundant middleware stack are active and a second selection of said plurality of layers of said redundant middleware stack are in standby (¶ 10);

means for managing a data replication partition shared between said primary node and said secondary node with data accessible to a selection of said plurality of layers of said active middleware stack, wherein said selection of said plurality of layers

of said active middleware stack correspond to said second selection of said plurality of layers of said redundant middleware stack in standby (¶ 4: both nodes may run identical software with identical configurations); and

means, responsive to detecting a failure at said primary node, transferring said virtual IP address from said primary node to said secondary node, remounting said data replication partition for access by said secondary node, activating said second selection of said plurality of layers of said redundant middleware stack which require access to said data in said data replication partition (¶ 38 lines 7-15, ¶ 6).

Gadir does not disclose:

turning turn off power to said primary node,

Cramer discloses a cluster computing system (figure 1) in which a secondary node may force a takeover of a primary node if the primary node is experiencing a fault (flowchart, figure 5). The secondary node sends a confirmation signal (step 513), and shuts down the primary node if it is not able to do so itself (step 519). Cramer discloses that this prevents resources from being used needlessly by the failed node after the second node has already taken over (column 2 lines 46-55). Gadir discloses that failover causes a disruption of service because of limited availability of services being available, and that lower priority services are eliminated when failover occurs to prevent disruptions (¶ 17). Using the negotiated shutdown system of Cramer would prevent the faulty node from disrupting the system by attempting to access resources after a secondary node has already taken over. Therefore, it would have been obvious to one

of ordinary skill in the art at the time of invention to incorporate the negotiated shutdown system of Cramer into the cluster computing system of Gadir, preventing loss of performance during failover.

As per claim 27, Gadir discloses:

The computer program product according to claim 25 for managing a high availability cluster, means for controlling said middleware stack and said redundant middleware stack comprising a selection from among a load balancer (¶ 12 last 3 lines), a web server (¶ 8 lines 5-7), a web application server (¶ 8 lines 5-7), a messaging server (¶ 1), a monitoring server (¶ 5 lines 1-2), and a database control server (¶ 1).

As per claim 28, Gadir discloses:

The computer program product according to claim 25 for managing a high availability cluster, further comprising: means for enabling a web application server active on said secondary node before failover points to access a messaging server and a database control server from said middleware stack of said primary node (¶ 32: slave nodes sent status information to master node through the network).

As per claim 29, Gadir discloses:

The computer program product according to claim 25 for managing a high availability cluster, further comprising: means for launching a message queue and a

relational database in said data replication partition (¶ 29 lines 12-15).

As per claim 30, Gadir does not disclose:

The computer program product according to claim 25 for managing a high availability cluster, further comprising: means, responsive to detecting said failure at said primary node, for sending a shutoff command to turn off power to said primary node.

Cramer discloses a cluster computing system (figure 1) in which a secondary node may force a shutdown of a primary node if the primary node is experiencing a fault (flowchart, figure 5). The secondary node sends a confirmation signal (step 513), and shuts down the primary node if it is not able to do so itself (step 519). Cramer discloses that this prevents resources from being used needlessly by the failed node after the second node has already taken over (column 2 lines 46-55). Gadir discloses that failover causes a disruption of service because of limited availability of services being available, and that lower priority services are eliminated when failover occurs to prevent disruptions (¶ 17). Using the negotiated shutdown system of Cramer would prevent the faulty node from distrupting the system by attempting to access resources after a secondary node has already taken over. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate the negotiated shutdown system of Cramer into the cluster computing system of Gadir, preventing loss of performance during failover.

As per claim 31, Gadir discloses:

The computer program product according to claim 25 for managing a high availability cluster, further comprising: means for monitoring a status of a plurality of services provided by said middleware stack; means, responsive to detecting a failure of a particular service from among said plurality of services, for restarting another instance of said particular service at said primary node with a particular persistent identifier (¶ 16: ports [service] are replaced when one is faulty, ¶ 38 lines 1-15: each port has a unique identification).

As per claim 32, Gadir discloses:

The computer program product according to claim 25 for managing a high availability cluster, further comprising: means for sending heartbeat requests between said primary node and said secondary node; and means, responsive to not receive a return of a heartbeat request, for detecting a failure in the non-responsive node (¶ 28 lines 7-15, ¶ 6).

Claims 2, 12, 16, and 24, 26, and 33 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gadir (US Patent 2003/0018927 A1) in view of Cramer (US Patent 6,920,580), and in further view of Microsoft Computer Dicitonary (fifth edition).

As per claim 2, Gadir discloses:

The system according to claim 1 for managing a high availability cluster, wherein said middleware stack and said redundant middleware stack support J2EE compliant web applications.

Microsoft Computer Dicitonary discloses that J2EE is a framework of systems designed for use in distributed applications that provides a services, protocols and functionality for such a system. Gadir discloses that his system is a clustered system with multiple nodes connected for constant communication (¶ 2), as does Cramer (column 2 lines 57-65). Using the J2EE framework would enable the nodes to be connected and in communication according to a set protocol. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate J2EE into the cluster systems of Gadir and Cramer, as a framework to provide services and functionality to the system.

As per claim 12, Gadir discloses:

The system according to claim 1 for managing a high availability cluster, further comprising: said middleware stack running in said primary node atop a Linux compliant operating system layer.

Microsoft Computer Dicitonary discloses that Linux is an operating system which is popular with network servers. The source code is open for modification, and provided free of charge. Gadir discloses that his system operates as a server on a network (¶ 1, ¶ 2), and that the primary and secondary nodes can run any operating system as long as they both use the same one (¶ 4 lines 7-11). Cramer discloses that his invention

operates as a network server as well (column 1 lines 15-17). The Linux operating system would be a free operating system which is commonly used in network server environments. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate the Linux operating system into the cluster systems of Gadir and Cramer, as an operating system to run the network servers.

As per claim 16, Gadir discloses:

The method according to claim 15 for managing a high availability cluster, further comprising: controlling said middleware stack and said redundant middleware stack to support J2EE compliant web applications.

Microsoft Computer Dicitonary discloses that J2EE is a framework of systems designed for use in distributed applications that provides a services, protocols and functionality for such a system. Gadir discloses that his system is a clustered system with multiple nodes connected for constant communication (¶ 2). Using the J2EE framework would enable the nodes to be connected and in communication according to a set protocol. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate J2EE into the cluster system of Gadir, as a framework to provide services and functionality to the system.

As per claim 24, Gadir discloses:

The method according to claim 15 for managing a high availability cluster, further comprising:

running said middleware stack in said primary node atop a Linux compliant operating system layer.

Microsoft Computer Dicitonary discloses that Linux is an operating system which is popular with network servers. The source code is open for modification, and provided free of charge. Gadir discloses that his system operates as a server on a network (¶ 1, ¶ 2), and that the primary and secondary nodes can run any operating system as long as they both use the same one (¶ 4 lines 7-11). Cramer discloses that his invention operates as a network server as well (column 1 lines 15-17). The Linux operating system would be a free operating system which is commonly used in network server environments. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate the Linux operating system into the cluster systems of Gadir and Cramer, as an operating system to run the network servers.

As per claim 26, does not Gadir disclose:

The computer program product according to claim 25 for managing a high availability cluster, furthering comprising: means for controlling said middleware stack and said redundant middleware stack to support J2EE compliant web applications.

Microsoft Computer Dicitonary discloses that J2EE is a framework of systems designed for use in distributed applications that provides a services, protocols and functionality for such a system. Gadir discloses that his system is a clustered system with multiple nodes connected for constant communication (¶ 2), as does Cramer (column 2 lines 57-65). Using the J2EE framework would enable the nodes to be connected and in communication according to a set protocol. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate J2EE into the cluster systems of Gadir and Cramer, as a framework to provide services and functionality to the system.

As per claim 33, Gadir does not disclose:

The computer program product according to claim 25 for managing a high availability cluster, further comprising: means for running said middleware stack in said primary node atop a Linux compliant operating system layer.

Microsoft Computer Dicitonary discloses that Linux is an operating system which is popular with network servers. The source code is open for modification, and provided free of charge. Gadir discloses that his system operates as a server on a network (¶ 1, ¶ 2), and that the primary and secondary nodes can run any operating system as long as they both use the same one (¶ 4 lines 7-11). Cramer discloses that his invention operates as a network server as well (column 1 lines 15-17). The Linux operating system would be a free operating system which is commonly used in network server

environments. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate the Linux operating system into the cluster systems of Gadir and Cramer, as an operating system to run the network servers.

Claim 14 is rejected under 35 U.S.C. 103(a) as being unpatentable over Gadir (US Patent 2003/0018927 A1) in view of Microsoft Computer Dicitonary (fifth edition).

Gadir discloses:

A system for managing J2EE compliant middleware in a high availability cluster during failover, comprising:

a first node (¶ 10: two or more nodes contain one or more virtual servers [layers], controlling resources and file systems. One node is the "master" node, while the rest are "slave" nodes);

a second node for running a partially active and partially standby redundant stack of J2EE compliant middleware (¶ 10);

a data replication partition shared between said first node and said second node (¶ 29 lines 12-15) enabling access to persistent data resources to only said first node, ¶ 13: each virtual server on a node manages its own resources and cannot "see" resources being managed by others); and

a heartbeat monitor for detecting a failure in said first node and transferring services provided by said first node to said second node, wherein said heartbeat monitor activates said partially standby redundant stack and remounts said data

replication partition for access by said activated partially standby redundant stack (¶ 38 lines 7-15, ¶ 6),

Gadir does not disclose:

[a first node for running an active stack of] J2EE compliant middleware; and

[a redundant stack of] J2EE compliant middleware

Microsoft Computer Dicitonary discloses that J2EE is a framework of systems designed for use in distributed applications that provides a services, protocols and functionality for such a system. Gadir discloses that his system is a clustered system with multiple nodes connected for constant communication (¶ 2). Using the J2EE framework would enable the nodes to be connected and in communication according to a set protocol. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to incorporate J2EE into the cluster system of Gadir, as a framework to provide services and functionality to the system.

Allowable Subject Matter

Claim 9 is objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Claim 9 states:

The system according to claim 1 for managing a high availability cluster, further comprising:

a first power supply dedicated to said primary node;

a second power supply dedicated to said secondary node;

first connection means for connecting said first power supply and said secondary node;

second connection means for connecting said second power supply and said primary node;

means, responsive to detecting said failure at said primary node, for sending a shutoff command via said first connection means for turn off power to said first power supply.

Claim 13 was examined and considered allowable over the prior art. Reasons for allowance of claim 13 are as follows:

Claim 13 contains the limitations:

a drbd data storage partition shared between said primary node and said secondary node, wherein a message queue and a database implemented by said active management queue controller and said active database controller of said primary node are activated at said drbd data storage partition, and

in response to detecting said failure at said primary node, said heartbeat monitor transfers said first virtual IP address from said active load balancing controller to said redundant load balancing controller and activates said redundant load balancing controller,

said heartbeat monitor activates and assigns said second virtual IP address to said redundant management queue controller and said redundant database controller, and said heartbeat monitor remounts said drbd partition for access by said redundant management queue controller and said redundant database controller.

The prior art does not teach a redundant cluster computing system in which a message queue and a database implemented by the active database controller of said primary node are activated at a common drbd data storage partition. While drbd storage means are known in the art, there is no teaching to activate and run a message queue and database of the primary node at a drbd storage partition, in the cluster configuration of claim 13.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Matt Urick whose telephone number is (571) 272-0805. The examiner can normally be reached on 8:00 - 4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Robert Beausoliel can be reached on (571) 272-3645. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

MTZ
***